4-2023

Trust Model System for the Energy Grid of Things Network Communications

Narmada Sonali Fernando
Portland State University, narmada@pdx.edu

Zhongkai Zheng
Portland State University, zhongkai@pdx.edu

John M. Acken
Portland State University, john.acken@pdx.edu

Robert B. Bass
Portland State University, rbass2@pdx.edu

Citation Details

N. S. Fernando, Z. Zeng, J. M. Acken and R. B. Bass, "Trust Model System for the Energy Grid of Things Network Communications," [POST-PRINT] 2023 IEEE Conference on Technologies for Sustainability (SusTech), Portland, OR, USA, 2023, pp. 280-287.

Trust Model System for the Energy Grid of Things Network Communications

N. Sonali Fernando, Zhongkai Zeng, John M. Acken, Robert B. Bass
Department of Electrical and Computer Engineering
Portland State University
Portland, Oregon, USA
narmada@pdx.edu, zhongkai@pdx.edu, acken@pdx.edu, robert.bass@pdx.edu

***Abstract*—Network communication is crucial in the Energy Grid of Things (EGoT). Without a network connection, the energy grid becomes just a power grid where the energy resources are available to the customer unidirectionally. A mechanism to analyze and optimize the energy usage of the grid can only happen through a medium, a communications network, that enables information exchange between the grid participants and the service provider. Security implementers of EGoT network communication take extraordinary measures to ensure the safety of the energy grid, a critical infrastructure, as well as the safety and privacy of the grid participants. With the dynamic nature of network communication of the EGoT, the information provided by the customer or the service provider can be falsified by a malicious attacker. Therefore, a trust model is necessary to monitor any abnormal activities. This paper describes a distributed trust model system that meets the need of the EGoT. This paper describes methods for evaluating and improving the distributed trust model using standard hypothesis testing metrics such as true positive, false positive, true negative, false negative, equal error rate, and F1 score. Example calculations are shown based on generated sample data.**

***Index Terms*—Trust Model, Security, Energy Services Interface, Distributed Energy Resources, Energy Grid of Things, Distributed Control Module, Trust**

I. Introduction

Network communication continues to gain popularity in many technologies involving people's lives, including the power grid. Without a network connection, the energy grid lacks interoperability and becomes a power grid. However, with many benefits resulting from expanding the energy grid, there are several significant concerns, namely the security and privacy of the grid participants and the security of power system critical infrastructure.

Two key ideas are necessary when implementing the security plan for an EGoT communication network. 1) use of the threat model, and 2) the three attributes of cyber security: confidentiality, integrity, and availability (CIA). The threat model helps understand potential threats to the communication network and evaluate whether the existing security measurements are sufficient. The implemented security features of the EGoT must abide by the CIA information security model.

Many publications about the security evaluation of energy grid, Distributed Energy Resources (DER), and communication network security exist [1]–[4]. In addition to securing network communications, some work has been done to instrument systems for security monitoring. Tonaboylu described a hardware system as "integrated with appropriate software that can monitor various parameters for various loads and sources that interconnect through the network" [5]. Monemi describes a cyber test bed to test "cyber attacks in an electrical power grid system with crucial components and data" [6]. Bahrami describes an economical circuit developed to warn users when communication is not encrypted [7]. The IEEE 2030.5 standard is a smart grid communications protocol that specifies information security via encryption. The work presented in this paper provides a mechanism to augment security of network communication system, specifically for the EGoT system's network communication via the addition of a Distributed Trust Model (DTM) System. In this paper, Section II provides a brief overview of trust models and characteristics published. Section III describes the DTM design. Section IV presents the implementation of the DTM System by describing the network communication methodology used in this study, as well as the messaging schema. Section V describes the evaluation methodology and sample plots generated using the DTM System simulator.

A. Threat Model for Communication

When developing any security solution, defining the security threat model is essential. For the DTM System described in this paper, the threat is to the information flow, not the actual power grid itself. Specifically, the threat is attacks that modify or interfere with the messages among the various actors within an EGoT. The DTM is not addressing passive eavesdropping attacks. The overall security of an EGoT addresses the traditional CIA security triad of confidentiality, integrity,

This work was performed under US DOE agreement DE-OE0000922.

and availability; these types of attacks on confidentiality are defended by implementing HTTPS and the security requirements of the IEEE 2030.5 standard. Integrity is protected by the encryption protocol specified in the standard. Additionally, the DTM System provides integrity checks by evaluating the contents and timing of various messages. Registrations and certification standards protect availability. The DTM System augments availability security by monitoring and evaluating the frequency and timing of the messages between various actors.

The threat models covered by the DTM consider several types of attackers. First are impostors pretending to be any of the DERs, Distributed Control Module (DCM) clients, or a DER Management System server. Second is any unreliable device at the Service Provisioning Customer (SPC) site. This unreliability can be intentional (as would be an attacker) or unintentional (such as equipment failures). Third are malicious devices at the SPC site due to computer virus infections. Fourth is a man-in-the-middle attack where messages on the internet are intercepted and modified. Fifth is a denial of service (DoS) attack based upon excessive frequency of messages. Sixth are unpredicted anomalies in the messages exchanged between actors. The DTM System examines the aggregate message threat at the Central Distributed Trust Aggregator (CDTA), not the individual DCMs. Power system reliability and efficiency will not be affected by a single misbehaving device but by many devices. Selecting what value to use for a significant number of devices is described later in the paper in the section about setting thresholds for alert messages. In summary, the threat model addressed by the DTM System is active message interference (modification, insertion, duplication, or blocking), not passive observation or eavesdropping.

B. EGoT basics

The Energy Grid of Things EGoT, shown in Figure 1, consist of actors such as Grid Operator (GO), whos responsibilities include procuring grid-DER services from Grid Service Provider (GSP) to maintain power system reliability and provide resilience. Figure 1 describes the EGoT ecosystem and the DTM System. The Energy Service Interface is a virtual point that governs the communication between the GSP and the SPC to conform to set expectations which are protection of privacy, provide privacy, develop trustworthiness, and affirm interoperability. The Grid operator, GSP, and CDTA are part of a Wide Area Network (WAN). The DCM, DER, and Distributed Trust Model Client (DTMC) are part of a Local Area Network (LAN).

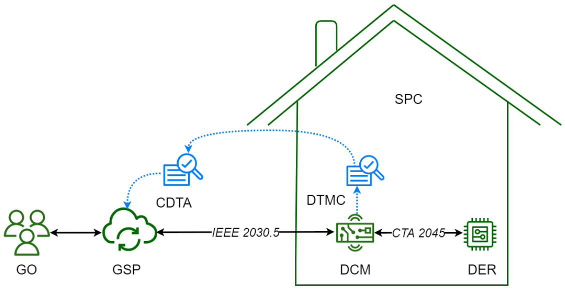

Fig. 1. The EGoT with the associate DTM System.

C. Goals of the DTM System

The DTM System continuously monitors and evaluates information exchange between actors and ensures security application of messages is on par with the CIA triad. To meet the confidentiality component of the CIA triad, the DTM monitors for any indicator of message content exposure to unauthorized parties. To meet the integrity component of the CIA triad, the DTM monitors for any sign of message altercation during transit or altercation done by an unauthorized party. Finally, to meet the accessibility component of the CIA triad, the DTM checks for any sign of message exposure to unauthorized parties.

The DTM does not determine the type of attack nor the exact cause of abnormalities. Instead, it evaluates the messages to identify any abnormalities and creates alerts. The DTM augments the existing security system to enhance grid security. It uses the current and historical trust data to evaluate and notify the appropriate level of response to the predetermined party. The DTM uses only the information provided to evaluate trust; it does not exert any direct control over any EGoT entity.

II. Background

A. Trust Models in general

A DTM improves the existing security of the grid. A DTM determines the trustworthiness of participants, protects customer privacy, and may or may not exert direct control based on the designer's intention (our intention is that it does not). One of the earliest trust-related publications provides reasoning why trust is missing in network communication security where, cryptography is limited to privacy, digital signature provides authenticity, and access control provides access to selective parties [8]. These authentications cannot verify abnormalities in the communication network [8].

The trust model survey conducted by [9] presents many types of trust model designs, such as distributed and centralized trust models. It is described by [9] that the distributed trust model has each trusted node calculate the trust scores compared to the centralized trust model where the central node calculates trust and trust nodes rely on the central trust node to get trust information. Trust is based on recommendation, where nodes rely on neighboring node recommendations before interacting with new nodes. In a direct trust-base system, each trusted agent calculates trust based on its experience with the interacting node. Trust model designs are flexible; [10] presents a hybrid trust calculation that uses both direct trust and recommendation.

Another hybrid trust module is presented by [11], who uses both feedback and self evaluation to calculate trust. Many publications use different methodologies to calculate trust, such as a vector base trust versus using a single trust score. Zhao and Li designed a vector-base trust system where the nodes use trust recommendations to obtain trust values of new nodes within the network [12].

Just like the implementation of trust models, the power grid and network communication security are essential to understand when designing a DTM for an EGoT. There are many publications relating to experiments conducted to understand network communication security. Tunaboylu et al. developed a hardware system that monitors interconnected parameters through a communication network [5]. Monemi et al. implemented a cyber test bed to conduct an electrical power grid cyber attack that impacted grid components and data [6]. Bahrami and Haisadeghi developed an economical circuit to inform users that the communication is not encrypted and exposes the network to vulnerabilities [7].

III. The Trust Model

A. Overall DTM System

The DTM System has two components: 1.) a DTMC located at the SPC and 2.) a CDTA located at the GSP. The responsibility of the DTMC is to perform a trust evaluation of the DCM message exchange between other actors (the GSP and the DER), maintain the results, and report the resulting evaluations to the CDTA. The DTMC has a Metric Vector of Trust (MVoT) that contains the trust evaluation of all the actors with the DCM exchange messages, including itself. The responsibility of the CDTA is to aggregate the trust evaluations provided by the DTMCs and perform analysis based on criteria set by the authoritative party, such as the GSP or grid operator. The set criteria are used to evaluate if any patterns of abnormalities are signs of threats.

B. DTM System Architecture

The DTM System block diagram is shown in Figure 2. The DTM System consists of two components, the DTMC and the CDTA. The DTMC, as mentioned earlier, is located at the SPC and connected to a LAN connection. The CDTA is located at the GSP and connected to a WAN connection. The DTMC contains a classifier, an MVoT, and an evaluator.

The classifier block, shown in Figure 2, evaluates the incoming messages to determine whether they are expected, unexpected, indeterminate, disconnect, error, or none. The classifier checks the message contents to determine message legitimacy. The classifier considers a message to be expected when all the necessary contents are present, and its initiation or response is in order instead of random. The classifier marks a message as an error if the message has a future time stamp, or the message fields are of the wrong data type. Other instances that result in an error classification are if the actor information is invalid or missing. If an actor randomly sends a response message without inquiry, the classifier identifies such action as unexpected. In many cases, the DTM cannot classify the incoming message, resulting in indeterminate classification. An example of an indeterminate classification is if an actor sends the same message twice. The classifier classifies a message to be disconnected if there are no messages from the actor within a predetermined time frame window. When there are no classifications needed, then that message is classified as none.

The evaluator block evaluates trust based on the message classification and the current MVoT values. Then, the evaluator block updates the MVoT variables with new values and sends updated values to the CDTA.

The MVoT is a vector of sixteen variables. Each variable is calculated by combining its current value and the data of the classified input file. MVoT values are collected and analyzed to detect abnormalities within the communication network. Having many variables helps to detect many different attacks instead of having few that show a particular actor is not trustworthy. For example, the MVoT variable *frequency of communication* possibly leads to identifying DoS attacks since it keeps track of the messaging frequency of an actor. The trust system can detect and alert the proper authority; at the same time, it can also determine just by looking at other MVoT variables, such as *certainty*, that there is enough data for the DTM System to decide if the right call is made about sending an alert to the authoritative party.

The CDTA is responsible for aggregating and analyzing all the MVoT data sent by every DTMC within

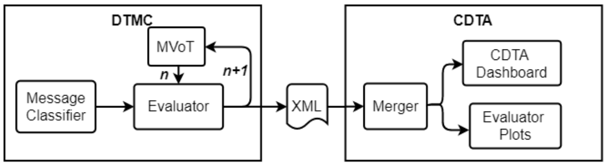

Fig. 2. Distributed Trust Model Architecture block diagram.

the EGoT system. The aggregation of MVoTs is made possible by the Merger block. Once the evaluator sends the updated MVoT variables, the Merger script at the CDTA merges the new values into the aggregated MVoT values.

The aggregated MVoT values are used to generate a dashboard for trust analysis and send recommendations and alerts to authoritative parties, such as the GSP and grid operator.

C. MVoT Calculations

The MVoT variables quantify different aspects of actors' trustworthiness, as derived from the message evaluation. The MVoTs for each actor are derived using multiple measures of evaluations.

1) **Certainty:** How certain is the DTM for each evaluation

$$C(i) = ((RFC) \times (1 - e^{(-\gamma \times TotMsg)}) \times (\frac{ComFreq}{Max\_ComFreq})) \quad (1)$$

Certainty has a significant influence on other MVoT variables. For example, the distrust and trust scores consider certainty when deriving their value instead of looking into the difference between the count of expected messages and the weighted unexpected messages. The MVoT variable certainty provides the confidence level to calculate a specific MVoT variable.

The certainty calculations have a weighted value, γ, which determines the rate of influence the total number of messages has on certainty. The specific term of the certainty equation, $(1 - e^{(-\gamma \times TotMsg)})$, represents the influence of increased total messages has on the certainty. Certainty also factors other MVoT variables, such as Relative Factor of Certainty (RFC), communication frequency, and time since the last communication. All these variables take into account the actor's interactions.

2) **Trust Score:** Overall trust score for each actor

$$TS = [CExMSG - (\alpha \times CUnMSG)] \times C \quad (2)$$

The trust score takes into account the count of expected messages from the actor, a weighted value alpha, and the count of unexpected messages. The weighted value α determines how much of an influence the unexpected messages, has on the overall trust score calculations. The trust score is derived from deducting the influence of unexpected messages from the expected messages and multiplying the result with the current certainty evaluation.

3) **Distrust Score:** Distrust score for each actor

$$DS = CUnMSG \times C \quad (3)$$

Having a separate distrust score clearly shows how untrustworthy any EGoT actor can be. The count of unexpected messages multiplied by the certainty provides the distrust score of an actor.

4) **Count Of Expected Messages:** Total count of expected messages for each actor.

$$CExMsg(i+1) = CExMsg(i) + 1 \quad (4)$$

5) **Count Of Unexpected Messages:** Total count of unexpected messages for each actor.

$$CUnMsg(i+1) = CUnMsg(i) + 1 \quad (5)$$

6) **Total Number Of Messages:** Count of total messages

$$TotMsg(i+1) = TotMsg(i) + 1 \quad (6)$$

Equation 4, the variable count of expected messages, keeps a count of all the messages classified as expected. Similarly, Equation 5, the variables count of unexpected messages. and Equation 6, the total number of messages, keep count of all the unexpected messages and of all the messages, respectively. The messages of these variables help other variables, such as the trust score, distrust score, and certainty, to calculate their values. Specifically, the trust score looks at the expected and unexpected messages, and certainty uses total messages when calculating its value.

7) **Time Stamp:** Time of the most recent message received from the actor

$$TimeStamp = time of actor's first message \quad (7)$$

The timestamp is used to understand the frequency of communication for each actor, which helps detect any abnormalities in communication rates. The variable time stamp helps to identify the timing information of an actor's first message and calculate the time since the last communication.

8) **Registration Date (Unix Time):** the first time a message is received from an actor.

Keeping track of the registration dates helps to understand the length of time an actor started sending messages. If an actor conducted an attack, having a record of the registration date indicates how long the actor participated in the grid and provides a time frame to look more closely for any other abnormalities during that time.

9) **Frequency of Communication:** How often an actor communicates with the DCM.

$$ComFreq = \frac{TotMsg}{CurrentTime - Regstr\_Time} \quad (8)$$

The communication frequency variable indicates the rate per unit times an actor communicates. Keeping track of the frequency of communication for an actor helps detect any drastic change in the communication frequency of an actor.

10) **Measured Transit Time:** The time difference for the message traveling from the source to the destination.

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{9}$$

11) **Average Transit Time:** Average expected transit time.

$$\mu_{n+1} = \frac{n \times \mu_n + x_{n+1}}{n+1} \tag{10}$$

Equation 9, keeps track of the time it takes for a message to reach the destination from the source for each actor. This variable data of Equation 10, helps derive the average message transit time. The measured transit time is compared against the average transit time to understand any irregularities.

12) **Time Since Last Communication:** Time Since Last Communication.

$$TSLC = Time\ delta\ of\ the\ last\ message\ received \tag{11}$$

It is essential to track how often an actor communicates; the time since the last communication variable keeps track to see if there is a significant irregularity in the rate at which an actor communicates.

13) **Standard Deviation Of Transit Time:** Extent of deviation for Transit time as a whole.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu_n)^2} \tag{12}$$

$$\sigma_{n+1} = \sqrt{\tfrac{n\times(\sigma_n)^2+(x_{i+1}-\mu_n)(X_{i+1}-\mu_{n+1})}{n+1}} \tag{13}$$

The standard deviation of transit time and the standard deviation of the transit time continuum are recalculated for each new message. This helps detect abnormalities in message transit time over a long period of time.

14) **Relative Factor of Certainty:** Certainty indicator of lean toward or against trust score or distrust score.

$$(|[CExMsg/(CExMsg + CUnMsg)] - 0.5|)x\beta \tag{14}$$

The relative factor of certainty variable uses the count of expected messages and the count of unexpected messages to quantify if an actor is trustworthy or leaning toward untrustworthy. β determines the maximum value of relative factor of frequency (RFC). This equation is defined such that the RFC does not exceed 50% of β. For example, if β is 1.2 then the RFC will not exceed 0.6.

15) **Count Of Timeouts:** Total count of timeouts for each actor (Protocol specific. E.g., 2030.5).

$$T\_Out(i+1) = T\_Out(i) + 1 \tag{15}$$

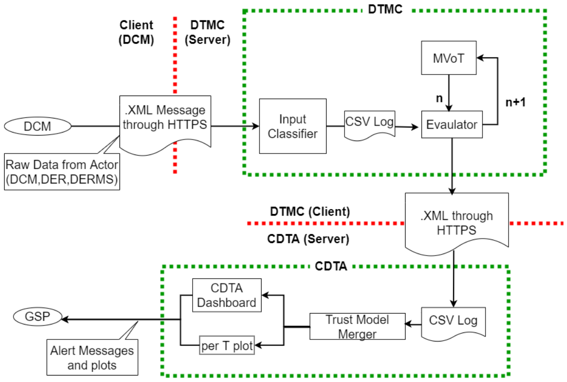

Fig. 3. DTM System network communication diagram.

The variable count of timeouts keeps track of an actor's timeouts. This gives a general idea of how often an actor has taken timeouts. This variable helps understand if an actor has an abnormal amount of timeouts.

16) **Count Of Alerts:** Total alerts sent out to each actor.

$$C\_Alrt(i+1) = C\_Alrt(i) + 1 \tag{16}$$

Keeping count of the number of alert messages sent helps understand if a specific actor or a group of actors continuously causes alerts to be sent out.

D. Alert Messages

The MVoT variables keep track of observed behaviors from the actor's messages. The CDTA collects and aggregates all the messages reported from many DTMCs. The CDTA analyzes the aggregated trust data and sends alert messages to authoritative parties such as the GSP. The decision to send messages is done by comparing one or several MVoT variables against predetermined threshold values. An alert is sent if an MVoT variable values exceed a threshold values.

IV. The DTM System

A. Network communication of the DTM System

We implemented a REST (Representational State Transfer) over Hypertext Transfer Protocol Secure (HTTPS) client/server information exchange system to send and receive messages between EGoT actors. The SSL (Secure Socket Layer) certificate methodology enables the security aspect of HTTPS.

The selection of HTTPS to send and receive data is a testimony to our dedication to staying true to the CIA triad. HTTPS conforms to the CIA triad. Recent experiments by [13] show it is an appropriate security solution to malicious attaches such as SQL injection attacks.

The DTMC receives encapsulated messages from the DCM over HTTPS. In this scenario, the DCM client

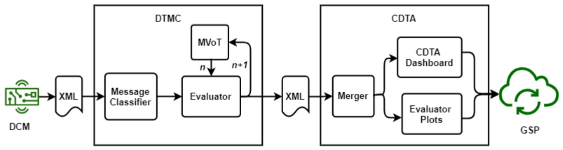

Fig. 4. Block diagram representing the DTM System component.

sends POST requests to the CDTA server. There is a REST API (Application Programming Interface) between the DTMC and the CDTA.

Figure 3 illustrates the trust information exchange between the EGoT actors and the client and server relationship when exchanging messages over HTTPS. As shown in Figure 3, the DCM sends encapsulated eXtensible Markup Language (XML) messages to the DTMC using an HTTPS client/server system; the DCM is the client, and the DTMC is the server. Then the DTMC sends the derived MVoT calculations to the CDTA via HTTPS. In this scenario, the client is the DTMC and the server is the CDTA.

B. Messaging schema

The DTMC server parses the encapsulated message into a Coma Separated Value (CSV) file, then sends it to the message classifier, which classifies the information. The classifier outputs a CSV file containing the following information: the Actor field contains either the message recipient or the message sender's name, and the message classification which evaluates the message content, a timestamp of when the message was sent, and message transit time.

- **Actor:** DCM, DER, GSP
- **Classification:** Expected, Unexpected, Indeterminate, Disconnect, Error, None
- **Time:** Message sent time
- **TX Time:** Message transit time

The classified message is an input to the evaluator along with the current MVoT entries of the associated actors whose messages were classified. The evaluator block uses these data to evaluate trust. The results are updated to the MVoT and sent to the CDTA. The DTMC message to the CDTA is in XML format.

The CDTA receives this message from the DTMC and converts the XML messages to CSV format. The merger script takes the latest trust evaluations and appends the received trust values to the aggregated list. The appended data are analyzed to detect major and minor abnormalities and report them to the authoritative party. Additionally, the CDTA dashboard of MVoT values is sent to the GSP for further analysis.

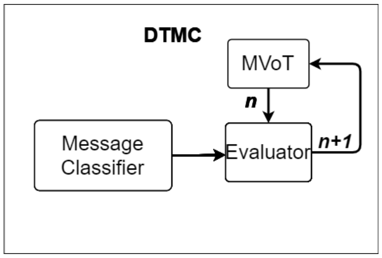

Fig. 5. The DTMC is programmed into the raspberry pi.

C. The implementation details (raspberry pi)

In our implementation, the DTMC is programmed within a Raspberry pi. We use a Raspberry Pi 4 Model B for the DTMC. All the programming modules inside the DTMC, such as the message classifier, evaluator, and MVoT, Figure 5, are programmed into the Raspberry Pi shown in Figure 6.

In our implementation, the DCM and the DTM both resides in the Raspberry Pi at the customer's home. The DCM communicates with the DTMC over a local host with isolated server client relationship. The Raspberry pi is connected to power, and a battery pack is included in case of a power loss.

D. Evolution of DTM System data:

This section describes the current status of our project and the evolution of DTM System data. To test the DTM System, we implemented a DTM System simulator with all the functionalities we plan to apply to the prototype. In place of a Message Classifier, a data generator script is used. The generated data contain the identical data that a classifier would produce, such as the Actor's name, message classification, message sent time, and transit

Fig. 6. The DTMC embedded into the raspberry pi.

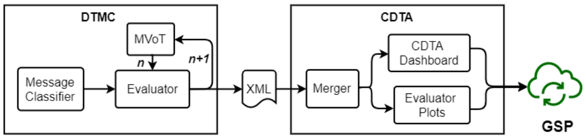

Fig. 7. Distributed Trust Model Simulator.

time. The remaining scripts were used as described in the Messaging Schema section.

The simulator helped to generate large amounts of data and analyze and evaluate the MVoT equations via the application of those generated data.

The generated data was applied to check hypothesis testing.

V. Trust Evaluation

The ultimate goal of the DTM System is to detect and send alerts about system abnormalities to a system authority, such as the GSP. The decoding technique for sending an alert is non-trivial. The DTM System accuracy rate must be high, otherwise an attack or an abnormality can go undetected. The CDTA sends alerts to the GSP by analyzing the aggregated MVoT variables provided by DTMCs. Two thresholds are set to check for any abnormalities in aggregated MVoT variables. Each MVoT variable has a predetermined value threshold, which is a maximum value that should not surpass the aggregated value. Any MVoT value surpassing the value threshold indicates an abnormality. The count threshold is a predetermined value. To send an alert, the count threshold is set to determine if many actors surpass the value threshold. In short, the value threshold helps identify whether the reported MVoT value is abnormal. The count threshold checks to see if there are a significant number of actors reporting abnormal MVoT variables and, if so, sends alerts to authoritative parties.

Our hypothesis testing uses a confusion matrix to determine DTM System accuracy in sending alerts. True positive is decided when the count of actors surpassing the value threshold is less than the count threshold, and there are no attacks. False positive is when the count of actors surpassing the value threshold is less than the count threshold even though an attack is present. False negative is determined when the count of actors surpassing the value threshold is greater than the count threshold, although there are no actual attacks. True negative is when the count of actors surpassing the value threshold is greater than the count threshold, and attacks are present.

Equations corresponding with the confusion matrix are called the *confusion metric equations*. Although there are many confusing metrics, our study uses a False Positive Rate (FPR), a false negative rate (FNR), and an F-1 score. Our research combines FPR and FNR to derive the balancing point, equal error rate (EER). The EER is the point where the FNR equals the FPR.

$$FPR = \frac{FP}{(FP + TN)} \tag{17}$$

$$FNR = \frac{FN}{(FN + TP)} \tag{18}$$

$$F1 = \frac{TP}{TP + 0.5(FP + FN)} \tag{19}$$

The FPR Equation 17 represents the fraction of false alerts sent by CDTA out of all the actual trustworthy events. The FNR Equation 18 represents the missed chance of not sending an alert out of all the untrustworthy events. FPR and FNR show a possible error rate of the CDTA decisions. Ideally, we want to have a minimum error rate. The EER is when the FPR is equal to FNR where one error rate is not greater or less than the other. Figure 8 shows the EER from generated data for a set value threshold of 900 seconds since the last communication MVoT variable when the system was under 12 attacks with an EER of 0.91. Where the value threshold is constant and the count of actors exceeding the count threshold varies. The x axis shows the variation in count threshold. Equation 19 represents the F-1 score, which is the harmonic mean of precision and sensitivity. Figure 9 shows the F-1 score for a set value threshold of 900 seconds since the last communication MVoT variable when the system was under 12 attacks with an EER of 0.91 and have the count threshold represented by x-axis varies.

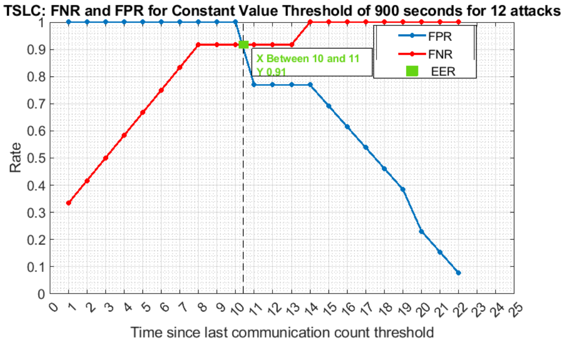

Fig. 8. Equal error rate example [14].

VI. Security-monitoring while protecting privacy

The DTM System provides added security for the communications between various actors within an EGoT network. As described above, a DTM System monitors messages and evaluates the contents to calculate the

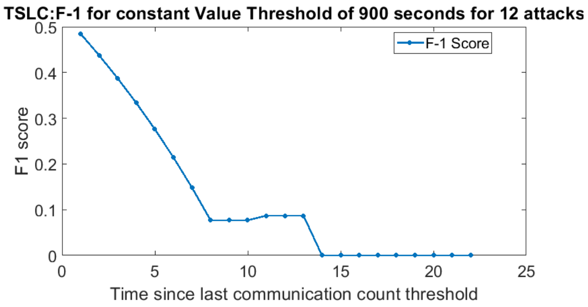

Fig. 9. F-1 Score example [14].

values that make up the MVoT. The evaluations and calculations require sufficient information to increase the reliability and efficiency of the power grid. However, for the goal of privacy, no customer information is gathered or saved that would compromise customer privacy. Furthermore, not collecting or keeping privacy-sensitive information prevents such information from intentional or accidental release.

VII. Summary and conclusions

The EGoT provides a mechanism to improve the reliability and efficiency of the electrical power grid. The EGoT uses standard communication security protocols for HTTPS and as specified in the IEEE 2030.5 standard. The DTM System provides additional security against security attacks that modify or interfere with the EGoT communication. The effectiveness of the implementation of the distributed trust was measured by applying standard statistical measures against generated date. The evaluation methodology and system are now available for data from the prototype system once it is in operation.

VIII. glossary

CDTA	Central Distributed Trust Aggregator
DCM	Distributed Control Module
DER	Distributed Energy Resources
DS	Distrust Score
DTM	Distributed Trust Model
DTMC	Distributed Trust Model Client
EGoT	Energy Grid of Things
EER	Equal Error Rate
ESI	Energy Service Interface
FN	False Negative
FNR	False Negative Rate
FPR	False Positive Rate
GO	Grid Operator
GSP	Grid Service Provider
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
LAN	Local Area Network
MVoT	Metric Vector of Trust
RFC	Relative Factor of Certainty
SDTT	Standard Deviation of Transit Time
SPC	Service Provisioning Customer
TS	Trust Score
TSLC	Time Since Last Communication
WAN	Wide Area Network
XML	eXtensible Markup Language

References

[1] C. Carter, I. Onunkwo, P. Cordeiro, and J. Johnson, "Cyber security assessment of distributed energy resources," in *2017 IEEE 44th Photovoltaic Specialist Conference (PVSC)*. IEEE, pp. 2135–2140. [Online]. Available: https://ieeexplore.ieee.org/document/8366503/

[2] Z. Zhu, Y. Hu, G. Xiao, and B. Zhou, "Cyber-physical security analysis of smart inverters under the pricing attacks," in *2021 China Automation Congress (CAC)*. IEEE, pp. 6205–6208. [Online]. Available: https://ieeexplore.ieee.org/document/9728459/

[3] O. T. Soyoye and K. C. Stefferud, "Cybersecurity risk assessment for california's smart inverter functions," in *2019 IEEE CyberPELS (CyberPELS)*. IEEE, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/8925257/

[4] Lidong Zhou and Z. Haas, "Securing ad hoc networks," vol. 13, no. 6, pp. 24–30. [Online]. Available: http://ieeexplore.ieee.org/document/806983/

[5] N. S. Tunaboylu, G. Shehu, M. Argin, and T. Yalcinoz, "Development of smart grid test-bed for electric power distribution system," in *2016 IEEE Conference on Technologies for Sustainability (SusTech)*. IEEE, pp. 184–187. [Online]. Available: http://ieeexplore.ieee.org/document/7897164/

[6] S. Monemi, D. Kamand, R. Thayi, S. Luong, and T. Venrick, "Smart grid cyber test bed development," in *2016 IEEE Conference on Technologies for Sustainability (SusTech)*. IEEE, pp. 102–108. [Online]. Available: http://ieeexplore.ieee.org/document/7897150/

[7] H. Bahrami and K. Hajsadeghi, "Circuit design to improve security of telecommunication devices," in *2015 IEEE Conference on Technologies for Sustainability (SusTech)*. IEEE, pp. 171–175. [Online]. Available: http://ieeexplore.ieee.org/document/7314342/

[8] A. Abdul-Rahman and S. Hailes, "A distributed trust model," in *Proceedings of the 1997 workshop on New security paradigms*, ser. NSPW '97. Association for Computing Machinery, pp. 48–60. [Online]. Available: https://doi.org/10.1145/283699.283739

[9] K. Govindan and P. Mohapatra, "Trust computations and trust dynamics in mobile adhoc networks: A survey," vol. 14, no. 2, pp. 279–298. [Online]. Available: http://ieeexplore.ieee.org/document/5770276/

[10] A. Abdul-Rahman and S. Hailes, "Supporting trust in virtual communities," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, vol. vol.1. IEEE Comput. Soc, p. 9. [Online]. Available: http://ieeexplore.ieee.org/document/926814/

[11] Li Xiong and Ling Liu, "PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities," vol. 16, no. 7, pp. 843–857. [Online]. Available: http://ieeexplore.ieee.org/document/1318566/

[12] H. Zhao and X. Li, "VectorTrust: Trust vector aggregation scheme for trust management in peer-to-peer networks."

[13] F. Wibowo, H. H. Nuha, and S. Wibowo, "Network security analysis using HTTPS with SSL on general election quick count website," in *2020 IEEE International Conference on Communication, Networks and Satellite (Comnetsat)*. IEEE, pp. 204–207. [Online]. Available: https://ieeexplore.ieee.org/document/9328940/

[14] N. Fernando, "The distributed trust model applied to the energy grid of things." [Online]. Available: https://pdxscholar.library.pdx.edu/open$_{a}ccess_{e}tds/5875$